SEGA: Variance Reduction via Gradient Sketching

Filip Hanzely[1] Konstantin Mishchenko [1] Peter Richtárik[1, 2, 3]

[1]KAUST [2]University of Edinburgh [3]Moscow Institute of Physics and Technology

Problem and Assumptions

Regularized Optimization

$$\min_{x\in\mathbb{R}^n} F(x) = f(x) + R(x) \tag{1}$$

- f: $\mathbf{M}$–smooth & μ–strongly convex convex:

$$f(x+h) \le f(x) + \langle \nabla f(x), h\rangle + \tfrac{1}{2}\langle \mathbf{M}h, h\rangle$$
$$f(x) + \langle \nabla f(x), h\rangle + \tfrac{\mu}{2}\|h\|^2 \le f(x+h)$$

(natural assumptions for ERM with linear predictors)

- R non-smooth, convex & proximable

New Oracle: Gradient Sketch

We do not have direct access to $\nabla f(x)$. Instead, we have access to a random linear transformation of the gradient:

$$\mathbf{S}^\top \nabla f(x) \in \mathbb{R}^b, \qquad \mathbf{S} \sim \mathcal{D} \tag{2}$$

- $\mathbf{S}$: random $n \times b$ matrix (b small)
- $\mathcal{D}$: distribution from which $\mathbf{S}$ is drawn

Goal

Design a proximal stochastic gradient-type method for solving (1) using the gradient sketch oracle (2).

Simple Algorithmic Idea

$$x^{k+1} = \mathrm{prox}_{\alpha R}(x^k - \alpha g^k), \tag{3}$$

α = stepsize; g^k = a "nice" estimator of $\nabla f(x^k)$.

How to design a good gradient estimator g^k?

Key Challenges:

- In the case when $\mathcal{D}$ is a distribution over standard basis vectors $e_1, \dots, e_n$ in $\mathbb{R}^n$, i.e., if we have access to random partial derivatives of f, then we can use

$$g^k = e_i^\top \nabla f(x^k) e_i,$$

and (3) reduces to proximal randomized coordinate descent (`CD`). However, `CD` does not work with non-separable regularizers R. So, we have an issue even in this simple case! How to resolve it?

- How to deal with gradient sketches coming from any distribution $\mathcal{D}$?

Resolution: The SEGA estimator. We will iteratively learn an unbiased variance-reduced estimator g^k of the gradient $\nabla f(x^k)$ by incorporating the latest information provided by the gradient sketch.

Constructing the SEGA Estimator

SEGA Estimator

1. Ask oracle for a gradient sketch at x^k: $\mathbf{S}_k^\top \nabla f(x^k)$
2. Define h^{k+1} as the closest (in some energy norm $\|h\|_{\mathbf{B}}^2 \overset{\text{def}}{=} h^\top \mathbf{B} h$, where $\mathbf{B} \succ 0$) vector to h^k consistent with the gradient sketch:

$$h^{k+1} = \arg\min_{h\in\mathbb{R}^n} \|h - h^k\|_{\mathbf{B}}^2 \quad \text{subject to} \quad \mathbf{S}_k^\top h = \mathbf{S}_k^\top \nabla f(x^k) \tag{4}$$

Closed-form solution of (4):

$$h^{k+1} = h^k + \mathbf{B}^{-1}\mathbf{Z}_k(\nabla f(x^k) - h^k); \qquad \mathbf{Z}_k \overset{\text{def}}{=} \mathbf{S}_k\left(\mathbf{S}_k^\top \mathbf{S}_k\right)^\dagger \mathbf{S}_k^\top$$

3. Define the `SEGA` estimator:

$$g^k = h^k + \theta_k \mathbf{B}^{-1}\mathbf{Z}_k(\nabla f(x^k) - h^k) \tag{5}$$

(θ_k is a random variable ensuring that g^k is unbiased)

Key property: As $x_k \to x^*$, we get $g^k \to 0$, and hence `SEGA` estimator is variance-reduced.

Variants:

- `biasSEGA` estimator: use h^{k+1} instead of g^k
- `subspaceSEGA` estimator: If $f(x) = \phi(\mathbf{A}x)$ for some matrix $\mathbf{A} \in \mathbb{R}^{d\times n}$, we can improve the `SEGA` estimator by exploiting the fact that ∇f lies in Range($\mathbf{A}^\top$). We do this by adding the constraint $h \in$ Range($\mathbf{A}^\top$) to (4).

SEGA (SkEtched GrAdient descent)

`SEGA` = Method (3) + `SEGA` estimator (5)
`biasSEGA` = Method (3) + `biasSEGA` estimator (4)
`subspaceSEGA` = Method (3) + `subspaceSEGA` estimator

Convergence of SEGA

Let $\mathcal{D}$ be the uniform distribution over standard basis vectors $e_1, \dots, e_n \in \mathbb{R}^n$, and choose $\mathbf{B} = \mathbf{I}$. Then with stepsize $\alpha = \Omega(\frac{1}{n\lambda_{\max}(\mathbf{M})})$ and some constant $\sigma > 0$, we have

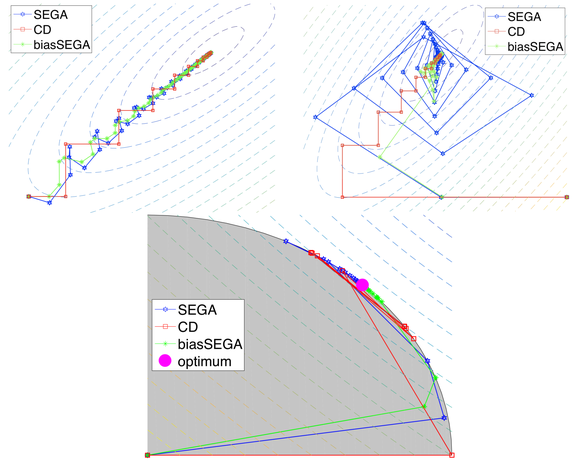
$$\mathbb{E}\left[\Phi^k\right] \le (1 - \alpha\mu)^k \mathbb{E}\left[\Phi^0\right],$$

where $\Phi^k \overset{\text{def}}{=} \|x^k - x^*\|^2 + \sigma\alpha\|h^k - \nabla f(x^*)\|^2$, $x^* = \arg\min_x F(x)$.

- Note that $x^k \to x^*$ and $h^k \to \nabla f(x^*)$
- General convergence result for any $\mathbf{B} \succ \mathbf{0}$ and any $\mathcal{D}$ can be found in the paper [1].
- `subspaceSEGA`: If $\mathcal{D}$ samples from the columns of $\mathbf{A}^\top$, the rate can be $\Omega(\frac{n}{d})$ faster than standard `SEGA`.
- For coordinate sketches, we designed an accelerated `SEGA`, and established accelerated rate (read next).

Iterates of SEGA (in 2D)

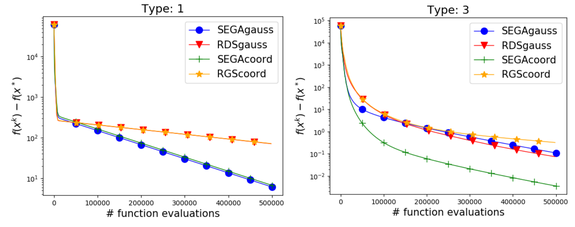
Iterates evolution of `SEGA`, `CD` and `biasSEGA` (updates made using h^{k+1} instead of g^k).



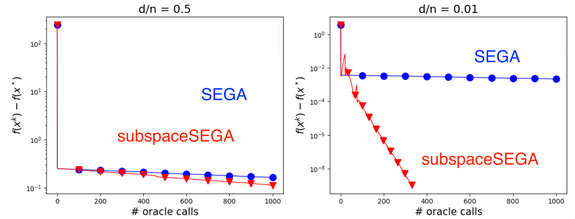
Bottom plot: R is the indicator function of the unit ball. While `CD` does not converge, `SEGA` does!

Experiments

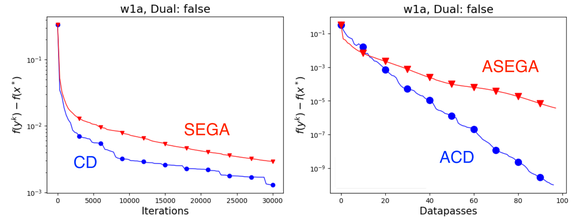
1. `SEGA` vs Random Direct Search (`RDS`) [2] (coordinate and Gaussian sketches) for derivative-free optimization



2. `SEGA` vs `subspaceSEGA`



3. `SEGA` vs Coordinate Descent (`CD`) [3] (left) and `ASEGA` vs Accelerated Coordinate Descent (`ACD`) [4, 5] (right) on ridge regression with $R = 0$



SEGA with Coordinate Sketches

Setup:

- $\mathbf{S}$ are column submatrices of the identity matrix
- Probability vector $p \in \mathbb{R}^n$: $p_i \overset{\text{def}}{=} \text{Prob}(e_i \in \mathbf{S})$
- Probability matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$: $\mathbf{P}_{ij} \overset{\text{def}}{=} \text{Prob}(e_i \in \mathbf{S}, e_j \in \mathbf{S})$
- ESO vector $v \in \mathbb{R}^n$ (for mini-batching) defined by:

$$\mathbf{P} \bullet \mathbf{M} \preceq \mathrm{Diag}(p \bullet v)$$

Acceleration: For coordinate sketches we also designed an accelerated variant of `SEGA`:

Algorithm Accelerated `SEGA` (`ASEGA`)

$x^0 = y^0 = z^0 \in \mathbb{R}^n$; $h^0 \in \mathbb{R}^n$; params $\alpha, \beta, \tau, \mu > 0$
for $k = 1, 2, \dots$ **do**
  $x^k = (1-\tau)y^{k-1} + \tau z^{k-1}$
  Sample $\mathbf{S}_k \sim \mathcal{D}$, and compute g^k and h^{k+1}
  $y^k = x^k - \alpha p^{-1} \bullet g^k$
  $z^k = \frac{1}{1+\beta\mu}(z^k + \beta\mu x^k - \beta g^k)$
end for

Rates: We prove the following iteration complexity bounds of `SEGA` and `ASEGA` with coordinate sketches:

Method	Complexity
`SEGA` importance sampling	$8.55 \cdot \frac{\mathrm{Tr}(\mathbf{M})}{\mu} \log\frac{1}{\epsilon}$
`SEGA` arbitrary sampling	$8.55 \cdot \left(\max_i \frac{v_i}{p_i\mu}\right) \log\frac{1}{\epsilon}$
`ASEGA` importance sampling	$9.8 \cdot \frac{\sum_i \sqrt{\mathbf{M}_{ii}}}{\sqrt{\mu}} \log\frac{1}{\epsilon}$
`ASEGA` arbitrary sampling	$9.8 \cdot \sqrt{\max_i \frac{v_i}{p_i^2\mu}} \log\frac{1}{\epsilon}$

Up to the constant factors 8.55 and 9.5, these rates are exactly the same as the rates of coordinate descent [3] and accelerated coordinate descent [4, 5]. So, we extend the reach of coordinate descent methods to problem (1) with a non-separable regularizer (e.g., arbitrary convex constraint)

References

[1] Filip Hanzely, Konstantin Mishchenko, and Peter Richtárik.
SEGA: Variance reduction via gradient sketching.
In *NeurIPS*, 2018.

[2] El Houcine Bergou, Peter Richtárik, and Eduard Gorbunov.
Stochastic three point method for minimizing nonconvex, convex and strongly convex functions.
Manuscript, 2018.

[3] Peter Richtárik and Martin Takáč.
On optimal probabilities in stochastic coordinate descent methods.
Optimization Letters, 10(6):1233–1243, 2016.

[4] Zeyuan Allen-Zhu, Zheng Qu, Peter Richtárik, and Yang Yuan.
Even faster accelerated coordinate descent using non-uniform sampling.
In *ICML*, 2016.

[5] Filip Hanzely and Peter Richtárik.
Accelerated coordinate descent with arbitrary sampling and best rates for minibatches.
arXiv:1809.09354, 2018.